LogisticRegression.R

Administrator

Thu Apr 07 12:02:46 2016

```
# logistic regression
# use when the response variable is discrete (0,1)
# predictor variable is continuous

# create ordered x values
xdat<- sort(runif(20,min=0,max=100))

# create y response data yes/no
ydat<- c(0,0,0,0,0,1,1,0,1,1,1,0,1,1,1,0,1,1,1,1)

# bind in a data frame
MyData <-data.frame(cbind(xdat,ydat))


# inspect data
head(MyData)
```

```
##        xdat ydat
## 1  6.431901    0
## 2  8.156290    0
## 3  8.745182    0
## 4 13.186337    0
## 5 15.267097    0
## 6 27.454373    1
```

```
# fit logistic regression model
MyModel<-glm(ydat~xdat, family = binomial("logit"), data=MyData)
summary(MyModel)
```

```
##
## Call:
## glm(formula = ydat ~ xdat, family = binomial("logit"), data = MyData)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.1686  -0.5725   0.2457   0.6651   1.3515
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -2.47422    1.30988  -1.889   0.0589 .
## xdat         0.07554    0.03311   2.281   0.0225 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
```

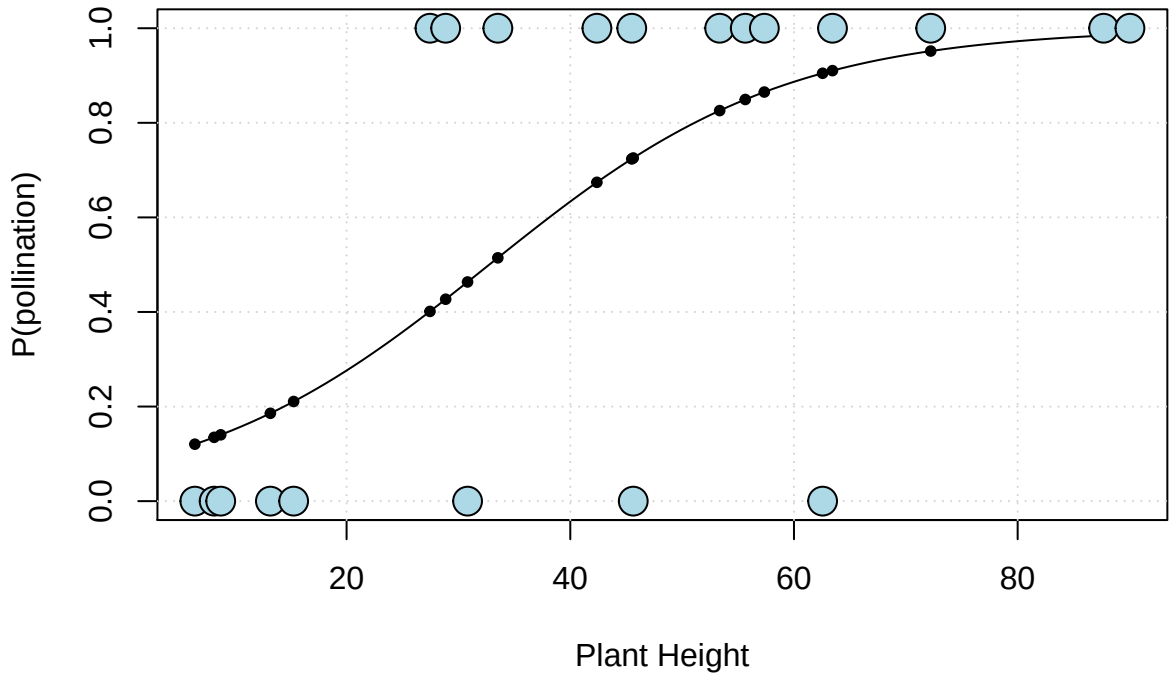
```
##     Null deviance: 26.920  on 19  degrees of freedom
## Residual deviance: 17.904  on 18  degrees of freedom
## AIC: 21.904
##
## Number of Fisher Scoring iterations: 5
```

```r
# create empty plot with grid
plot (x=xdat,y=ydat,xlab="Plant Height", ylab="P(pollination)",type="n",ylim=c(0,1))
grid()

# add curve of predicted values
curve(predict(MyModel,data.frame(xdat=x),type="resp"),add=TRUE)

# add predicted points
points(x=xdat,y=fitted(MyModel),pch=20)

# add observed points
points(x=xdat,y=ydat,cex=2,pch=21,bg="lightblue")
```




```r
# Different data structure with successes and failures

myX <- sort(runif(5,min=0,max=100))
Successes <- c(10,20,1,75,0)
Failures <- c(2,5,1,100,200)
CountData <- data.frame(myX,Successes,Failures)
```

```r
# fit logistic regression model
MyModel<-glm(cbind(Successes,Failures)~myX, family = binomial("logit"), data=CountData)
summary(MyModel)
```

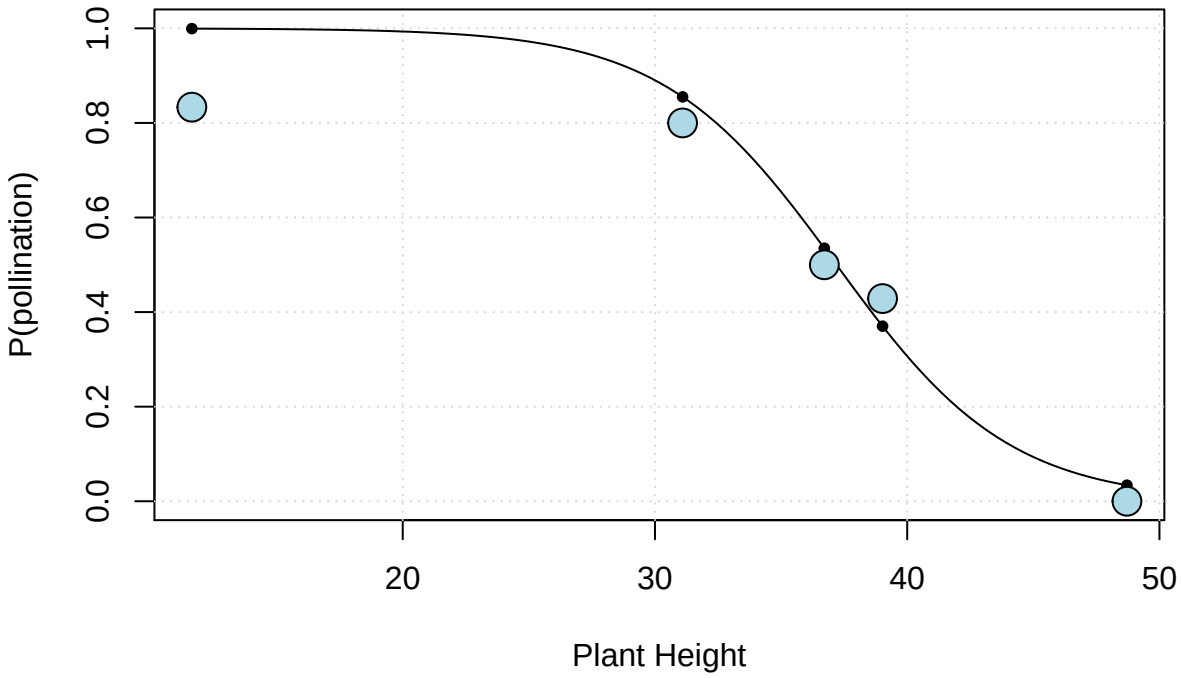
```
##
## Call:
## glm(formula = cbind(Successes, Failures) ~ myX, family = binomial("logit"),
##     data = CountData)
##
## Deviance Residuals:
##       1        2        3        4        5
## -4.3525  -0.7474  -0.0989   1.5845  -3.7132
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) 10.82261    1.42192   7.611 2.71e-14 ***
## myX         -0.29096    0.03559  -8.175 2.95e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 193.402  on 4  degrees of freedom
## Residual deviance:  35.811  on 3  degrees of freedom
## AIC: 52.49
##
## Number of Fisher Scoring iterations: 7
```

```r
# create empty plot with grid
plot (x=myX,y=(Successes/(Successes+Failures)),
      xlab="Plant Height", ylab="P(pollination)",
      type="n",ylim=c(0,1))
grid()

# add curve of predicted values
curve(predict(MyModel,data.frame(myX=x),type="resp"),add=TRUE)

# add predicted points
points(x=myX,y=fitted(MyModel),pch=20)

# add observed points
points(x=myX,y=(Successes/(Successes+Failures)),cex=2,pch=21,bg="lightblue")
```


P(pollination)
0.0
0.2
0.4
0.6
0.8
1.0
20
30
40
50
Plant Height